UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/809,873	03/25/2004	Wolfgang Theilmann	13909-161001	7587

32864 7590 09/21/2007
FISH & RICHARDSON, P.C.
PO BOX 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER	
AHLUWALIA, NAVNEET K	
ART UNIT	PAPER NUMBER
2166	

MAIL DATE	DELIVERY MODE
09/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/809,873	THEILMANN ET AL.	
	Examiner	Art Unit	
	Navneet K. Ahluwalia	2166	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 August 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-14 and 16-28* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-14 and 16-28* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *02/26/07, 03/01/2007*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 08/07/2007 has been entered.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to

consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

4. Claims 1, 3 – 14, 16 – 28 rejected under 35 U.S.C. 103(a) as being unpatentable over Theilmann et al. ('Theilmann' herein after) (US 2004/0126750 A1) further in view of Beavers et al. ('Beavers' herein after) (US 2004/0002049 A1).

With respect to claim 1,

Theilmann discloses a method, performed by one or more processing devices, for use in an electronic learning system that stores information as learning objects, the method comprising: designating a target learning object as a project object and storing version dependency data in the project object, the version dependency data identifying at least a version of a first object upon which the project object directly depends, and a version of a second object upon which the project object indirectly depends; wherein the first object stores dependency data identifying the second object upon which the first object depends and wherein the first object does not store version dependency data identifying the version of the second object upon which the first object depends (paragraphs 7 – 11, 39 – 42 and 101).

Theilmann however doesnot disclose the storing of the versions and their dependency explicitly as claimed.

Beavers however teaches the storing of the versions and their dependency as claimed in paragraphs 149 and 174, Beavers.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because there are directed towards the same field of invention of electronic learning systems. Furthermore, the detailed stored information regarding the versions and their dependency and other metadata would make the flow of the information easily understood and remove unnecessary duplication (paragraphs 174 – 177, Beavers).

With respect to claim 3,

Theilmann as modified discloses the method of claim 1, wherein designating comprises storing data in the project object that indicates that the target learning object is the project object (paragraph 101, Theilmann).

With respect to claim 4,

Theilmann as modified discloses the method of claim 1, wherein the target learning object comprises a portal to other learning objects in the electronic learning system (paragraphs 53 – 59, Theilmann).

With respect to claim 5,

Theilmann as modified discloses the method of claim 1, wherein the other learning objects define a course offered via the electronic learning system (paragraphs 25 – 27, Theilmann).

With respect to claim 6,

Theilmann as modified discloses the method of claim 4, wherein the target learning object comprises a glossary of a course (paragraphs 25 – 27, Theilmann).

With respect to claim 7,

Theilmann as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: identifying learning objects upon which the project object depends; moving the project object and learning objects upon which the project object depends between the local repository and the master repository (paragraphs 63 – 65, Theilmann).

With respect to claim 8,

Theilmann as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: copying the version of the first object from the master repository to the local repository without copying the project object to the local repository; and resolving dependencies associated with the version of the first object in accordance with a predefined rule (paragraphs 70 – 72, 83 and 89, Theilmann).

With respect to claim 9,

Theilmann as modified discloses the method of claim 8, wherein the version of the first object depends on the second object, and resolving comprises making the version of the first object depend on a most current version of the second object in the local repository (paragraphs 91 – 93, Theilmann).

With respect to claim 10,

Theilmann as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: copying the project object, the version of the first object, and the version of the second object from the master repository to the local repository; creating a second version of the first object; and updating the version dependency data in the project object to reference the second version of the first object (paragraphs 70 – 72, 83 and 89 – 93, Theilmann).

With respect to claim 11,

Theilmann as modified discloses the method of claim 1, wherein at least one of the first and second objects stores information about a dependent object (paragraph 108, Theilmann).

With respect to claim 12,

Theilmann as modified discloses the method of claim 11, wherein the information comprises an identity of the dependent object (paragraph 108, Theilmann).

With respect to claim 13,

Theilmann as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: copying the version of the first object from the master repository to the local repository without copying the project object to the local repository; and resolving dependencies associated with the version of the first object in favor of current versions of objects on which the first object depends (paragraphs 70 – 72, 83 and 89 – 93, Theilmann).

With respect to claim 14,

Theilmann discloses a computer program product for use in an electronic learning system that stores information as learning objects, the computer program product being tangibly embodied in an information carrier, the computer program product being operable to cause one or more machines to: designate a target learning object as a project object; store version dependency data in the project object, the version dependency data identifying at least a version of a first object upon which the

project object directly depends, and a version of a second object upon which the project object indirectly depends; wherein the first object stores dependency data identifying the second object upon which the first object depends, and wherein the first object doesnot store version dependency data identifying the version of the second object upon which the first object depends (paragraphs 7 – 11, 39 – 42 and 101, Theilmann).

Theilmann however doesnot disclose the storing of the versions and their dependency explicitly as claimed.

Beavers however teaches the storing of the versions and their dependency as claimed in paragraphs 149 and 174, Beavers.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because there are directed towards the same field of invention of electronic learning systems. Furthermore, the detailed stored information regarding the versions and their dependency and other metadata would make the flow of the information easily understood and remove unnecessary duplication (paragraphs 174 – 177, Beavers).

With respect to claim 16,

Theilmann as modified discloses the computer program product of claim 14, wherein designating comprises storing data in the project object that indicates that the target learning object is the project object (paragraph 101, Theilmann).

With respect to claim 17,

Theilmann as modified discloses the computer program product of claim 14, wherein the target learning object comprises a portal to other learning objects in the electronic learning system (paragraphs 53 – 59, Theilmann).

With respect to claim 18,

Theilmann as modified discloses the computer program product of claim 14, wherein the other learning objects define a course offered via the electronic learning system (paragraphs 25 – 27, Theilmann).

With respect to claim 19,

Theilmann as modified discloses the computer program product of claim 14, wherein the target learning object comprises a glossary of a course (paragraphs 25 – 27, Theilmann).

With respect to claim 20,

Theilmann as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions operable to cause the one or more machines to: identify learning objects upon which the project object depends; move the project object and learning objects upon which the project object depends between the local repository and the master repository

(paragraphs 63 – 65, Theilmann).

With respect to claim 21,

Theilmann as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions operable to cause the one or more machines to: copy the version of the first object from the master repository to the local repository without copying the project object to the local repository; and resolve dependencies associated with the version of the first object in accordance with a predefined rule (paragraphs 70 – 72, 83 and 89, Theilmann).

With respect to claim 22,

Theilmann as modified discloses the computer program product of claim 14, wherein the version of the first object depends on the second object, and resolving comprises making the version of the first object depend on a most current version of the second object in the local repository (paragraphs 91 – 93, Theilmann).

With respect to claim 23,

Theilmann as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available

learning objects, and the computer program product further comprises instructions operable to cause the one or more machines to: copy the project object, the version of the first object, and the version of the second object from the master repository to the local repository; create a second version of the first object; and update the version dependency data in the project object to reference the second version of the first object (paragraphs 70 – 72, 83 and 89 – 93, Theilmann).

With respect to claim 24,

Theilmann as modified discloses the computer program product of claim 14, wherein at least one of the first and second objects stores information about a dependent object (paragraph 108, Theilmann).

With respect to claim 25,

Theilmann as modified discloses the computer program product of claim 14, wherein the information comprises an identity of the dependent object (paragraph 108, Theilmann).

With respect to claim 26,

Theilmann as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions

cause the one or more machines to: copy the version of the first object from the master repository to the local repository without copying the project object to the local repository; and resolve dependencies associated with the version of the first object in favor of current versions of objects on which the first object depends (paragraphs 70 – 72, 83 and 89 – 93, Theilmann).

With respect to claim 27,

Theilmann as modified discloses the method of claim 1, wherein the version of the first object and the version of the second object store object dependency data but not version dependency data, wherein the object dependency data for the version of the first object identifies one or more first learning objects upon which the version of the first object depends but does not identify versions of the one or more first learning objects, and wherein object dependency data for the version of the second object identifies one or more second learning objects upon which the version of the second object depends but does not identify versions of the one or more second learning objects (paragraphs 7 – 11, 39 – 42 and 101).

With respect to claim 28,

Theilmann as modified discloses the computer program product of claim 14, wherein the version of the first object and the version of the second object store object dependency data but not version dependency data, wherein the object dependency data for the version of the first object identifies one or more first learning objects upon

which the version of the first object depends but does not identify versions of the one or more first learning objects, and wherein object dependency data for the version of the second object identifies one or more second learning objects upon which the version of the second object depends but does not identify versions of the one or more second learning objects (paragraphs 7 – 11, 39 – 42 and 101, Theilmann).

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

7. Claims 1, 3 – 14, 16 – 28 rejected under 35 U.S.C. 103(a) as being unpatentable over Cook et al. ('Cook' herein after) (US 2002/0168621 A1) further in view of Goodman et al. ('Goodman' herein after) (US 2006/0059253 A1).

With respect to claim 1,

Cook discloses a method, performed by one or more processing devices, for use in an electronic learning system that stores information as learning objects, the method comprising: designating a target learning object as a project object (paragraph 0077, Cook); storing version dependency data in the project object, the version dependency data identifying at least a version of a first object upon which the project object directly depends (paragraph 0078, Cook), and a version of a second object upon which the project object indirectly depends (paragraphs 0079 – 0081, Cook) wherein the version of the first object depends on the version of the second object (paragraphs 0091 – 0092, Cook).

Cook does not explicitly disclose the version dependency data as claimed.

Goodman however teaches the object and data dependencies in paragraphs 0316 and 0378. Goodman furthermore teaches version control in paragraphs 0138, 0170 and 0173.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because the version checking of Goodman's method would minimize the updating problems of Cook's method (paragraph 0014, Goodman).

Claims 3 – 13, 27 are rejected under the same rationale given for claim 1. The citations of the elements claimed and taught are listed below.

With respect to claim 3,

Cook as modified discloses the method of claim 1, wherein designating comprises storing data in the project object that indicates that the target learning object is the project object (paragraphs 0098 – 0099, Cook).

With respect to claim 4,

Cook as modified discloses the method of claim 1, wherein the target learning object comprises a portal to other learning objects in the electronic learning system (paragraphs 0099 – 0100, Cook).

With respect to claim 5,

Cook as modified discloses the method of claim 1, wherein the other learning objects define a course offered via the electronic learning system (paragraph 0079, Cook).

With respect to claim 6,

Cook as modified discloses the method of claim 4, wherein the target learning object comprises a glossary of a course (paragraph 0081, Cook).

With respect to claim 7,

Cook as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: identifying learning objects upon which the project object depends, moving the project object and learning objects upon which the project object depends between the local repository and the master repository (paragraphs 0104 – 0106, Cook).

With respect to claim 8,

Cook as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: copying the version of the first object from the master repository to the local repository without copying the project object to the local repository and resolving dependencies associated with the version of the first object in accordance with a predefined rule (paragraphs 0140 – 0142, Cook).

With respect to claim 9,

Cook as modified discloses the method of claim 8, wherein the version of the first object depends on the second object, and resolving comprises making the version of

the first object depend on a most current version of the second object in the local repository (paragraph 0104, Cook).

With respect to claim 10,

Cook as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: copying the project object, the version of the first object, and the version of the second object from the master repository to the local repository (paragraph 0104, Cook); creating a second version of the first object; and updating the version dependency data in the project object to reference the second version of the first object (paragraphs 0140 – 0142 , Cook).

With respect to claim 11,

Cook as modified discloses the method of claim 1, wherein at least one of the first and second objects stores information about a dependent object (paragraphs 0283 – 0285, Cook).

With respect to claim 12,

Cook as modified discloses the method of claim 11, wherein the information comprises an identity of the dependent object (paragraph 0285, Cook).

With respect to claim 13,

Cook as modified discloses the method of claim 1, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the method further comprises: copying the version of the first object from the master repository to the local repository without copying the project object to the local repository (paragraph 0104, Cook); and resolving dependencies associated with the version of the first object in favor of current versions of objects on which the first object depends (paragraph 0110, Cook).

With respect to claim 14,

Cook discloses a computer program product for use in an electronic learning system that stores information as learning objects, the computer program product being tangibly embodied in an information carrier, the computer program product being operable to cause one or more machines to: designate a target learning object as a project object (paragraph 0077, Cook); store version dependency data in the project object, the version dependency data identifying at least a version of a first object upon which the project object (paragraph 0078, Cook), and a version of a second object upon which the project object indirectly depends (paragraphs 0079 – 0081, Cook) wherein the version of the first object depends on the version of the second object (paragraphs 0091 – 0092, Cook).

Cook does not explicitly disclose the version dependency data as claimed.

Goodman however teaches the object and data dependencies in paragraphs 0316 and 0378. Goodman furthermore teaches version control in paragraphs 0138, 0170 and 0173.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because the version checking of Goodman's method would minimize the updating problems of Cook's method (paragraph 0014, Goodman).

Claims 16 – 26 and 28 are rejected under the same rationale given for claim 14. The citations of the elements claimed and taught are listed below.

With respect to claim 16,

Cook as modified discloses the computer program product of claim 14, wherein designating comprises storing data in the project object that indicates that the target learning object is the project object (paragraphs 0098 – 0099, Cook).

With respect to claim 17,

Cook as modified discloses the computer program product of claim 14, wherein the target learning object comprises a portal to other learning objects in the electronic learning system (paragraphs 0099 – 0100, Cook).

With respect to claim 18,

Cook as modified discloses the computer program product of claim 14, wherein the other learning objects define a course offered via the electronic learning system (paragraph 0079, Cook).

With respect to claim 19,

Cook as modified discloses the computer program product of claim 14, wherein the target learning object comprises a glossary of a course (paragraph 0081, Cook).

With respect to claim 20,

Cook as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions operable to cause the one or more machines to: identify learning objects upon which the project object depends, move the project object and learning objects upon which the project object depends between the local repository and the master repository (paragraphs 0104 – 0106, Cook).

With respect to claim 21,

Cook as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions operable to cause the one or more machines to: copy the version of the first object from the master repository to the local repository without copying the project object to the local repository and resolve dependencies associated with the version of the first object in accordance with a predefined rule (paragraphs 0140 – 0142, Cook).

With respect to claim 22,

Cook as modified discloses the computer program product of claim 14, wherein the version of the first object depends on the second object, and resolving comprises making the version of the first object depend on a most current version of the second object in the local repository (paragraph 0104, Cook).

With respect to claim 23,

Cook as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions operable to cause the one or more machines to: copy the project object, the version of the first object, and the version of the second object from the master repository to the local

repository (paragraph 0104, Cook); create a second version of the first object; and update the version dependency data in the project object to reference the second version of the first object (paragraphs 0140 – 0142 , Cook).

With respect to claim 24,

Cook as modified discloses the computer program product of claim 14, wherein at least one of the first and second objects stores information about a dependent object (paragraphs 0283 – 0285, Cook).

With respect to claim 25,

Cook as modified discloses the computer program product of claim 14, wherein the information comprises an identity of the dependent object (paragraph 0285, Cook).

With respect to claim 26,

Cook as modified discloses the computer program product of claim 14, wherein the electronic learning system comprises a master repository that stores globally-available learning objects and a local repository that stores locally-available learning objects, and the computer program product further comprises instructions to cause the one or more machines to: copy the version of the first object from the master repository to the local repository without copying the project object to the local repository (paragraph 0104, Cook); and resolve dependencies associated with the version of the

first object in favor of current versions of objects on which the first object depends (paragraph 0110, Cook).

With respect to claim 27,

Cook as modified discloses the method of claim 1, wherein the version of the first object and the version of the second object store object dependency data but not version dependency data, wherein the object dependency data for the version of the first object identifies one or more first learning objects upon which the version of the first object depends but does not identify versions of the one or more first learning objects, and wherein object dependency data for the version of the second object identifies one or more second learning objects upon which the version of the second object depends but does not identify versions of the one or more second learning objects (paragraph 0316, Goodman).

With respect to claim 28,

Cook as modified discloses the computer program product of claim 14, wherein the version of the first object and the version of the second object store object dependency data but not version dependency data, wherein the object dependency data for the version of the first object identifies one or more first learning objects upon which the version of the first object depends but does not identify versions of the one or more first learning objects, and wherein object dependency data for the version of the second object identifies one or more second learning objects upon which the version of

the second object depends but does not identify versions of the one or more second learning objects (paragraph 0316, Goodman).

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Navneet K. Ahluwalia whose telephone number is 571-272-5636.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Alam T. Hosain can be reached on 571-272-3978. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Navneet

Navneet K. Ahluwalia
Examiner
Art Unit 2166

Dated: 09/15/2007

SPE 2169